Spring源码编译教程

下载Spring源码所需要的依赖

下载Spring源码

修改IDEA的gradle配置

运行代码

问题1

问题2

问题3

作者: 周瑜

此教程是基于周瑜老师的**Spring5.3.10注释版源码**编译的,并不是Github原生的Spring5.3.10源码,有一些差别,但都是gradle配置文件的微小改动,比如把某些依赖从optional改成compile级别(主要是为了方便编译),其他都没改动。

讲的是最新的Spring版本,我之前用的是2019的某个IDEA版本,但是我尝试过后发现编译不成功,所以改用了最新版本IDEA版本2021.1.3,所以对于大家而言也尽量用这个版本,用其他IDEA版本可能会遇到各种各样的问题,解决问题会比较费时间,所以为了节省大家和我的时间,请大家用2021.1.3这个IDEA版本。

Spring带注释源码地址:

git clone的地址为: https://gitee.com/archguide/spring-framework-5.3.10.git

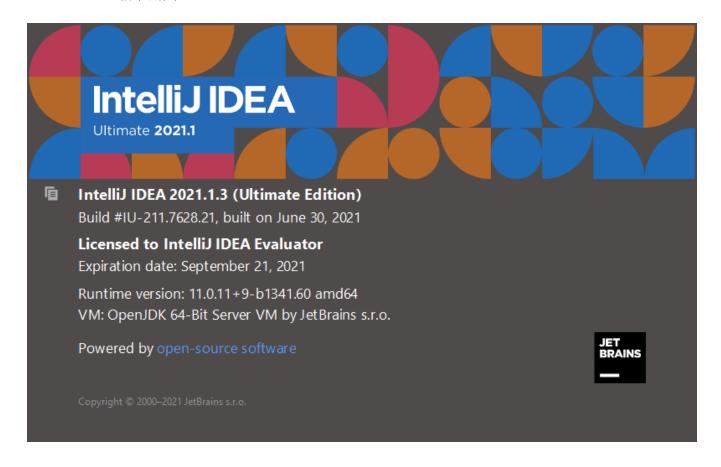
附上2021.1.3版的百度网盘链接:

链接: https://pan.baidu.com/s/1X79-2bFGtkL0763QjAya3w 提取码: uk7w

此链接中还有IDEA破解所需要的工具包,和一个.gradle.zip压缩包(后续会用到,我是用的360压缩软件进行压缩的,建议大家也用这个软件来解压,有同学反馈用其他软件可能解压会遇到问题,上面网盘链接里有一个.gradle的压缩包和一个未压缩的.gradle文件夹),还有一个JDK1.8的安装包(因为如果用稍微老一点的1.8小版本,也会出现奇葩问题,所以也尽量用我提供的这个JDK)

附上IDEA破解教程链接: https://www.exception.site/essay/idea-reset-eval

2021.1.3IDEA版本截图:



下载Spring源码所需要的依赖

百度网盘链接: https://pan.baidu.com/s/1X79-2bFGtkL0763QjAya3w 提取码: uk7w

下载得到.gradle.zip压缩包,并解压,比如解压到D盘



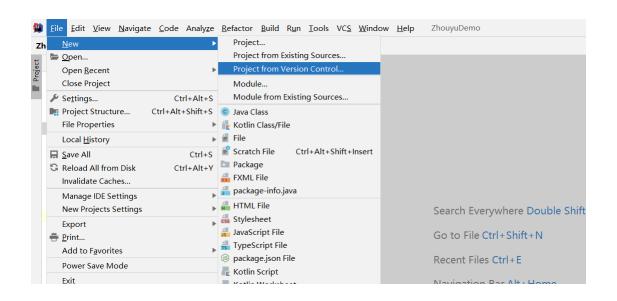
因为Spring源码存在很多依赖包,如果大家自行下载,会需要下很久(1小时都有可能),所以我直接把 我电脑上的依赖包给到大家。

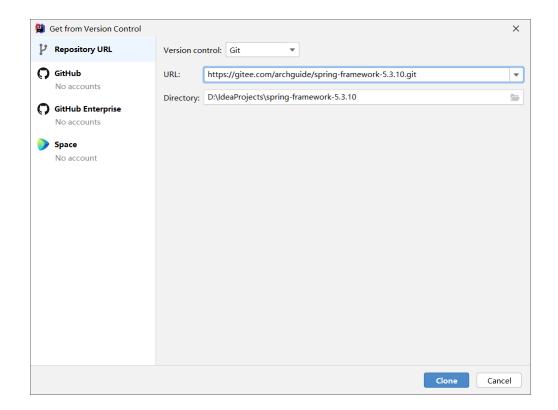
Spring是通过gradle来编译源码下载依赖的,.gradle文件夹可以理解为gradle的仓库(和mave类似,不懂gradle的先这么理解),而我给大家的这个仓库,只包含了Spring源码所需要的依赖。

下载Spring源码

git clone的地址为: https://gitee.com/archguide/spring-framework-5.3.10.git

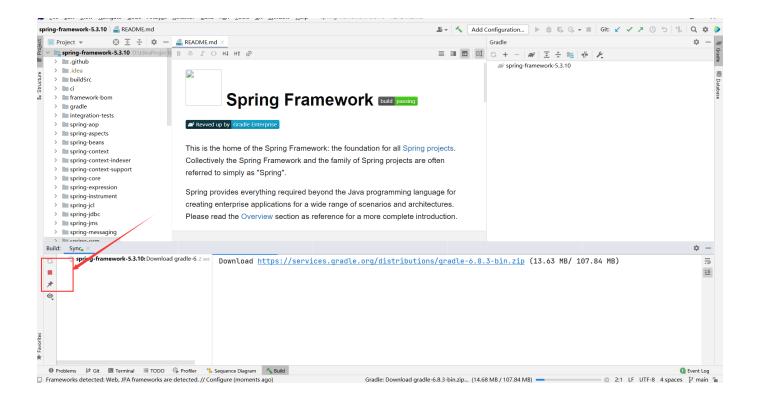
建议直接用IDEA的git来下载源码:



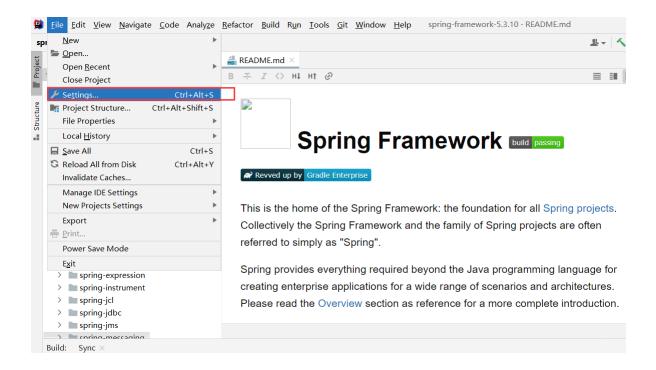


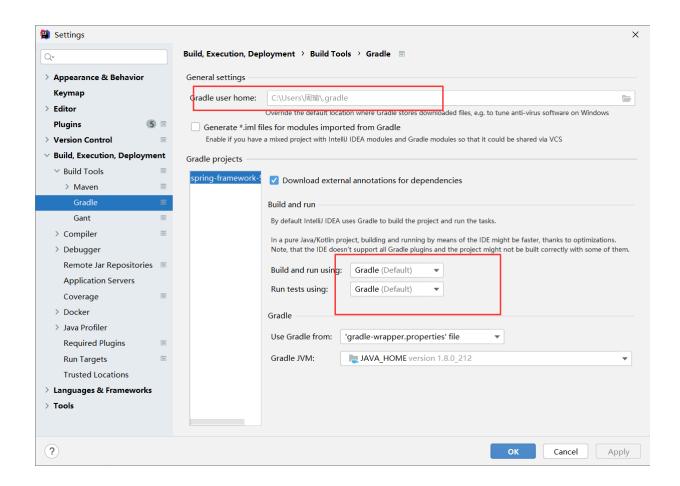
输入地址,点击Clone,就会开始下载源码工程(因为是从gitee上下载,所以会比从github上下载快很多)。

一旦下载完成,IDEA就会自动下载gradle,下载完gradle就会开始下载Spring源码依赖,但是我们已经有现成的了,所以可以直接取消这个过程。

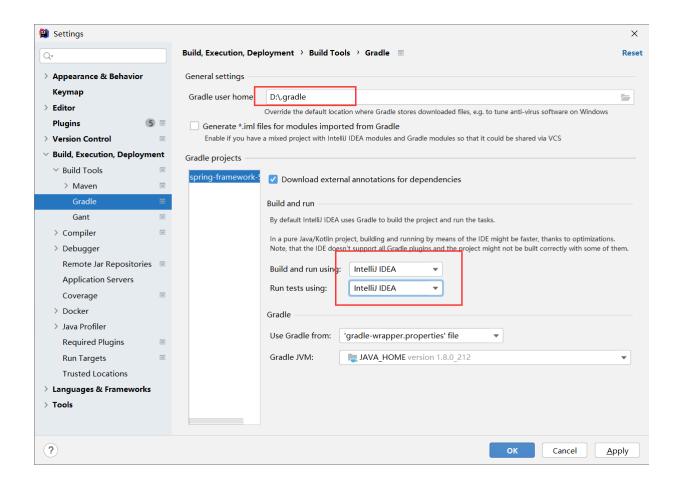


修改IDEA的gradle配置

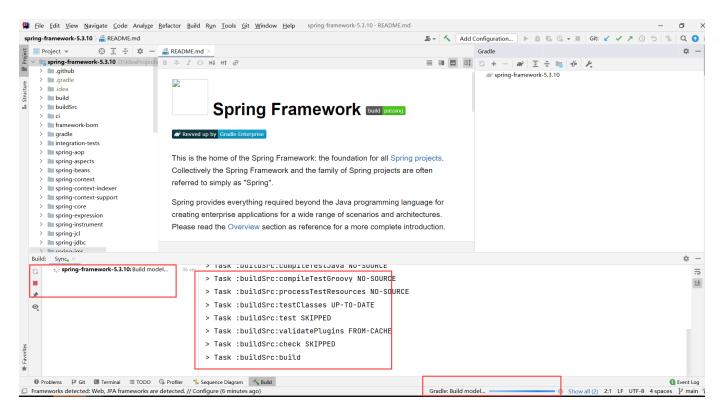




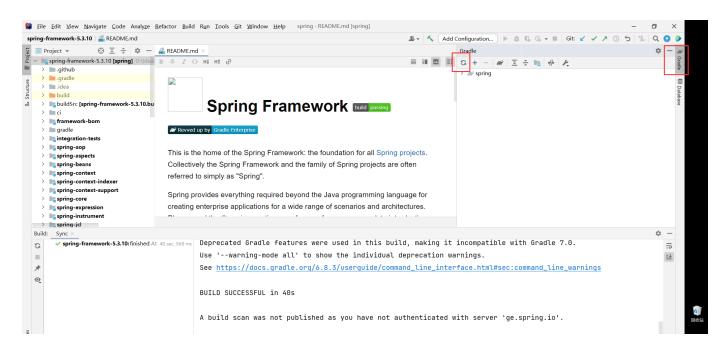
首先把gradle user home改为.gradle压缩包的解压之后的文件路径,比如D:\.gradle 然后把Build and run suing和Run tests using都改为IntelliJ IDEA,其他都不用动,改为之后如下图:



改完之后点击Apply,再点击OK,会自动触发gradle的重新编译。

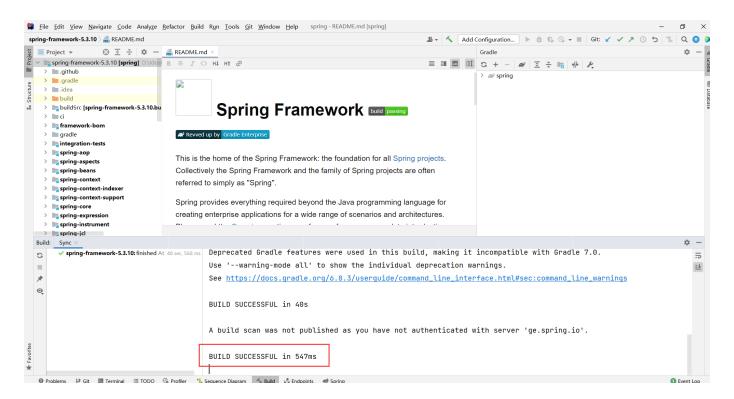


如果没有触发可以,点击



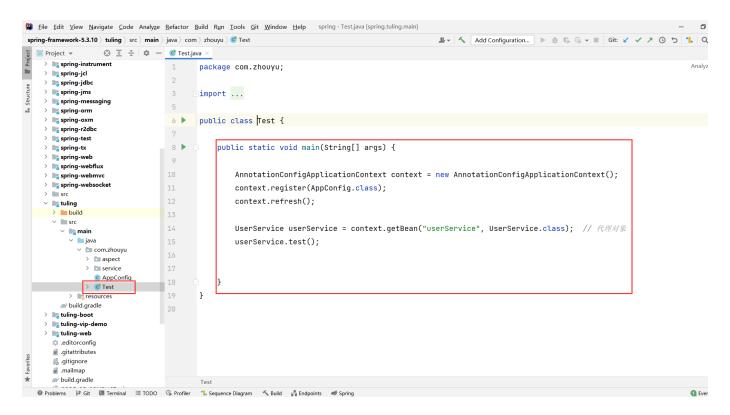
正常情况下,此时gradle编译将比较快,会有一个索引文件过程,但是不需要额外的下载gradle和依赖了。

编译成功截图:



运行代码

编译成功后,在左侧可以看到如下模块,其中有一个tuling模块,这是我写的一个模块,可以直接运行,在它下面有一个Test类,直接运行main方法。



问题1

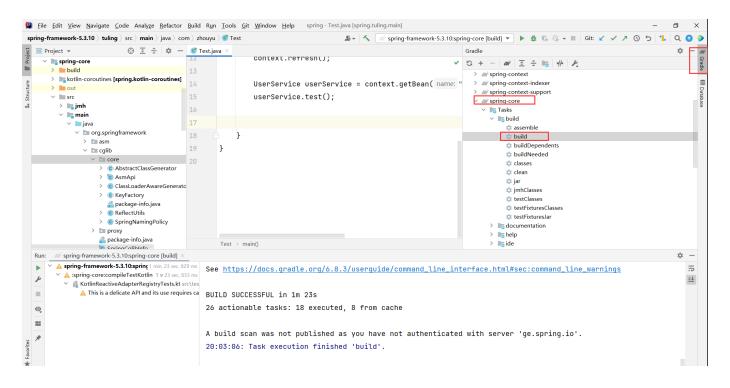
第一次运行可能会比较慢,在运行过程也可能会出现问题,比如

```
🚇 File Edit View Navigate Code Analyze Refactor Build Run Tools Git Window Help spring - Test.java [spring.tuling.main]
                                                                                                                           単マ | 🍕 | Test ▼ | ▶ 黄 😘 🚱 ▼ 🔳 | Git: 🗸 ✓ メ 🔘 与 👢 Q 🕡 🕨
 \textbf{spring-framework-5.3.10} \ \rangle \ \textbf{tuling} \ \rangle \ \textbf{src} \ \rangle \ \textbf{main} \ \rangle \ \textbf{java} \ \rangle \ \textbf{com} \ \rangle \ \textbf{zhouyu} \ \rangle \ \textbf{@} \ \textbf{Test}
                           > Di core
                                                6
                                                       public class Test {
                 > 🛅 lang
                 > 🖿 objenesis
Structure
                 > 🖿 util
            > kotlin
                                               8
                                                             public static void main(String[] args) {
            > resources
                                             9
                                                                  AnnotationConfigApplicationContext context = new AnnotationConfigApplicationContext();
          > lim testFixtures
                                             11
        context.register(AppConfig.class);
       spring-expression
spring-instrument
                                                                  context.refresh():
       spring-jcl
       spring-jdbc
                                                                  UserService userService = context.getBean( name: "userService", UserService.class); // 代理对象
       spring-messaging
                                                                  userService.test();
                                               16
     > 📭 spring-oxm
       spring-r2dbo
       spring-test
       spring-tx
  Build: Sync × Build Output
   spring: build failed At 2021/8/22 19:59 with 8 sec, 643 ms
                                                    D:\IdeaProjects\spring-framework-5.3.10\spring-core\src\main\java\org\springframework\cglib\core\AbstractClassGe
      ▲ Kotlin: Language version 1.5 is experimental, there ar
                                                                                                                                                                                                =+
                                                    java: 程序包org.springframework.cglib.core.internal不存在
      A Kotlin: Classpath entry points to a non-existent locative
      ▲ Kotlin: Classpath entry points to a non-existent location
      AbstractClassGenerator.iava.spring-cr

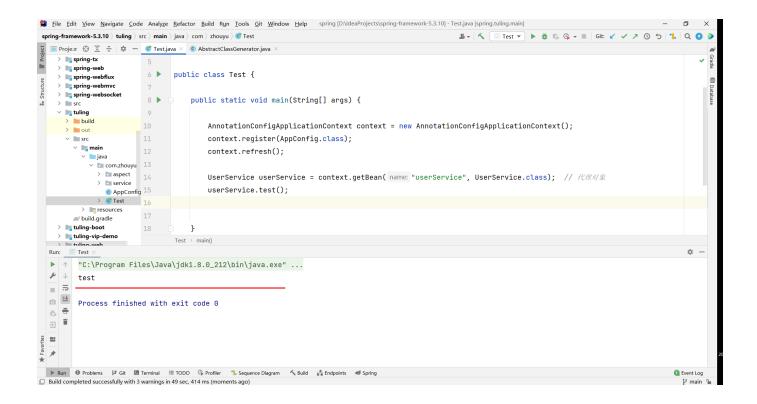
₱ 程序包org.springframework.cglib.core.internal不存

        ● 程序包org.springframework.cglib.core.internal不存● 找不到符号:37
   0
        47
投不到符号:47
        49
投不到符号:49
        4 找不到符号:176
        战不到符号:204
            ☐ Build completed with 100 errors and 3 warnings in 8 sec, 643 ms (a minute ago)
                                                                                                                                                                      17:1 LF UTF-8 Tab* € 1 main 1
```

那么请运行一下:



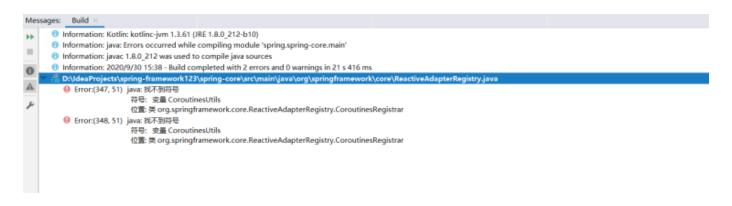
再次执行Test类中的main方法,可能就直接运行成功了:



到此,恭喜你,你已经成功的编译好了Spring源码,可以直接查看并进行调试了。

问题2

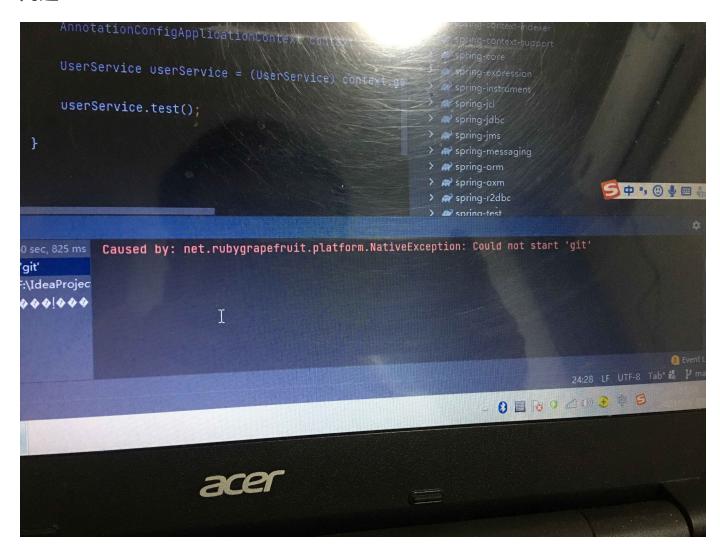
如果出现了:



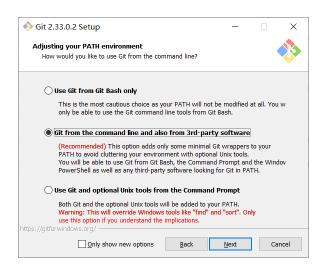
报错的CoroutinesUtils是一个kotlin中的类,解决办法:

点击File -> Project Structure -> Libraries -> "+" -> Java, 然后选择spring-framework/spring-core/kotlin-coroutines/build/libs/kotlin-coroutines-5.2.4.BUILD-SNAPSHOT.jar, 在弹出的对话框中选择**spring-core.main**, 在重新运行Test类中的main方法即可,注意我图中是报错的模块spring-core, 所以操作的是spring-core.main, 如果是其他模块报类似的错,就做类似的操作。

问题3



需要重新安装电脑上的git,并且最好是按照最新版本的git(上面网盘中有按照文件),安装的时候注意以下页面选择第二项:



如果大家在编译的过程中遇到了其他问题,那么可以在本文后面进行评论,最好是把解决方案也给出,方便其他小伙伴,感谢。